

J1050 U.S. PTO  
10/028320  
12/28/01

### **CLAIM FOR CONVENTION PRIORITY**

Assistant Commissioner for Patents  
Washington, D.C. 20231

**Sir:**

The benefit of the filing date of the following prior foreign application in the following foreign country is hereby requested, and the right of priority provided in 35 U.S.C. § 119 is hereby claimed:

Japanese Patent Application No. 2001-192670

**Filed: June 26, 2001**

In support of this claim, enclosed is a certified copy of said prior foreign application. Said prior foreign application was referred to in the oath or declaration. Acknowledgment of receipt of the certified copy is requested.

Respectfully submitted,

**BURNS, DOANE, SWECKER & MATHIS, L.L.P.**

**Date:** December 28, 2001

By: William Chawand, RN 30882, for  
Platon N. Mandros  
Registration No. 22,124

P.O. Box 1404  
Alexandria, Virginia 22313-1404  
(703) 836-6620

日 本 国 特 許 庁  
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出 願 年 月 日

Date of Application:

2001年 6月26日

出 願 番 号

Application Number:

特願2001-192670

出 願 人

Applicant(s):

三菱電機株式会社

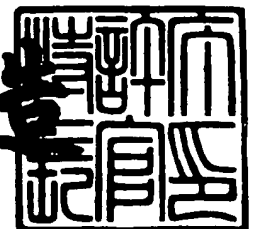


CERTIFIED COPY OF  
PRIORITY DOCUMENT

2001年 7月27日

特 許 庁 長 官  
Commissioner,  
Japan Patent Office

及 川 耕 造



【書類名】 特許願

【整理番号】 531742JP01

【提出日】 平成13年 6月26日

【あて先】 特許庁長官 殿

【国際特許分類】 H04N 7/30

【発明者】

【住所又は居所】 東京都千代田区丸の内二丁目2番3号 三菱電機株式会社  
社内

【氏名】 谷口 征浩

【特許出願人】

【識別番号】 000006013

【氏名又は名称】 三菱電機株式会社

【代理人】

【識別番号】 100102439

【弁理士】

【氏名又は名称】 宮田 金雄

【選任した代理人】

【識別番号】 100092462

【弁理士】

【氏名又は名称】 高瀬 彌平

【手数料の表示】

【予納台帳番号】 011394

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【プルーフの要否】 不要

【書類名】 明細書

【発明の名称】 ビットモデリングの処理方法及びそれを用いる処理回路

【特許請求の範囲】

【請求項1】 J P E G 2 0 0 0 のビットモデリングのシグニフィカンスプロパゲーションデコーディングパス ( S i g n i f i c a n c e p r o p a g a t i o n d e c o d i n g p a s s ) の処理方法において、

処理対象であるビット及び周囲のビット群の有意フラグの状態に応じて変化するデータのコンテキスト及びデシジョンと、

前記処理対象であるビット及び周囲のビット群の符号ビットの状態に応じて変化する符号ビットのコンテキスト及びデシジョンとを同時に生成しておき、

前記処理対象であるビットの値が1のときのみ前記符号ビットのコンテキスト及びデシジョンを採用し、前記有意フラグを更新し、

前記処理対象であるビットの値が0のときは廃棄し、

前記処理対象であるビットの値が1か0に係わらず、処理済フラグを更新するビットモデリングの処理回路を、

同一グループの4個のビットに同時に適用し、並列に処理することを特徴とするビットモデリングの処理方法。

【請求項2】 J P E G 2 0 0 0 のビットモデリングのマグニチュードリファインメントパス ( M a g n i t u d e r e f i n e m e n t p a s s ) の処理方法において、

処理対象であるビットが前記マグニチュードリファインメントパスで初めて処理されるかどうかの情報である有意第2ビットと処理済フラグ及び有意フラグとを参照して、処理するかどうかを判断し、

前記マグニチュードリファインメントパスで処理する場合は、前記処理対象であるビットのコンテキスト及びデシジョンを生成し、前記処理済フラグも更新するビットモデリングの処理回路を、

同一グループの4個のビットに同時に適用し、並列に処理することを特徴とするビットモデリングの処理方法。

【請求項3】 J P E G 2 0 0 0 のビットモデリングのクリーンアップパス

(Cleanup pass) の処理方法において、

処理対象である同一グループのビットが全て未処理のとき、まとめて処理できるかどうかを判断し、

前記同一グループのビット群が全て非有意のときは特別なコンテキスト及びデシジョンを生成する前記ビットモデリングの第 1 の処理回路と、

処理済のビットについては何も行わず、

非有意のビットについて処理を行う前記ビットモデリングの第 2 の処理回路とを設け、

前記第 1 の処理回路の 1 つと、前記同一グループの 4 個のビットに前記第 2 の処理回路とを同時に適用し、並列に処理することを特徴とするビットモデリングの処理方法。

【請求項 4】 J P E G 2 0 0 0 のビットモデリングの処理方法において、同一ビットプレーンについて、3 種類のコーディングパスを順番に処理することを特徴とするビットモデリングの処理方法。

【請求項 5】 J P E G 2 0 0 0 のビットモデリングの処理方法において、同一ビットプレーンの隣り合った 3 個のグループを、3 種類のコーディングパスごとに並列に処理することを特徴とするビットモデリングの処理方法。

【請求項 6】 請求項 5 記載のビットモデリングの処理方法において、同一ビットプレーンで複数並列に処理することを特徴とするビットモデリングの処理方法。

【請求項 7】 請求項 5 または 6 記載のいずれかのビットモデリングの処理方法において、

複数のビットプレーンで並列に処理することを特徴とするビットモデリングの処理方法。

【請求項 8】 請求項 1 記載のビットモデリングの処理方法を用いる処理回路において、

処理対象であるビットのデータの値を格納するレジスタと、

処理対象であるビット及び周囲のビット群の有意フラグ及び符号ビットを格納するレジスタと、

前記処理対象であるビットの処理済フラグを格納するレジスタとを備えることを特徴とする処理回路。

【請求項 9】 請求項 2 記載のビットモデリングの処理方法を用いる処理回路において、

処理対象であるビットのデータの値を格納するレジスタと、

処理対象であるビット及び周囲のビット群の有意フラグを格納するレジスタと

処理対象であるビットがマグニチュードリファインメントパスで初めて処理されるかどうかの情報である有意第 2 ビットを格納するレジスタとを備えることを特徴とする処理回路。

【請求項 1 0】 請求項 5、6、8 記載のいずれかのビットモデリングの処理方法を用いる処理回路において、

コードブロックサイズ分のデータビット、符号ビット、処理済フラグ、有意フラグ及び有意第 2 ビットとを格納するレジスタを備えることを特徴とする処理回路。

【請求項 1 1】 請求項 5、6、8 記載のいずれかのビットモデリングの処理方法を用いる処理回路において、

データビット、符号ビット、処理済フラグ、有意フラグ及び有意第 2 ビットとを処理の対象とするビット分だけを格納するレジスタを備えることを特徴とする処理回路。

【発明の詳細な説明】

【0 0 0 1】

【発明の属する技術分野】

この発明は、ビットモデリングの処理方法及びそれを用いる処理回路に係り、特に J P E G 2 0 0 0 の符号化におけるビットモデリングの高速化に関するビットモデリングの処理方法及びそれを用いる処理回路である。

【0 0 0 2】

【従来の技術】

J P E G 2 0 0 0 ( I S O / I E C F D I S 1 5 4 4 4 - 1 で定義。以下、

JPEG 2000と示す。)の符号化処理は、量子化係数を符号ビット及び絶対値で表す。コードブロックという単位で絶対値を表すビットを上位ビットプレーンから下位ビットプレーンの順に各ビットの上下左右の関係を調べながらコンテキスト及びデシジョンを生成する(以下、ビットモデリングと示す。)

## 【0003】

また、コードブロック内の絶対値を表すビットプレーンの最上位から全てのビットが0であるビットプレーンがN0個続くときは、別に処理する。

## 【0004】

例えば、絶対値としてMビットプレーンの精度があり、あるコードブロックにおいて、全ての絶対値の上位N0ビットが0であるようなときは、 $N = M - N0$ ビットプレーンのみビットモデリングの対象となる。

## 【0005】

ビットモデリングの対象となる最上位のビットプレーンをMSB (Most Significant Bit プレーン、以下、MSBと示す。)、最下位のビットプレーンをLSB (Less Significant Bit プレーン、以下、LSBと示す。)と呼ぶ。

## 【0006】

コードブロック内のデータは、MSBからLSBまでビットプレーンごとに処理して行く。各ビットプレーンにおいては縦方向に4個のビットをまとめてグループ化しておき、ラスタ順にビットモデリングを行う。

## 【0007】

図14は横8(0~7)×縦12(0~11)のコードブロックサイズでビットモデリングを行うときのグループを処理する順番(0~23)を示す。

## 【0008】

また、図15はビットプレーンのグループを跨ぐ個々のビット(縦方向に4個ずつのビット、例えば、横0×縦4(0~3)、横1×縦4(0~3))の処理順(0~7)を示す。

## 【0009】

ビットモデリングにはSignificance propagation

decoding pass (以下、sigパスと示す。)、Magnitude refinement pass (以下、refパスと示す。)、Cleanup pass (以下、clnパスと示す。)という3種のコーディングパスがあり、sigパス、refパス、clnパスの順番で処理される。

## 【0010】

即ち、この処理は、あるビットプレーンにおいて図14、15に示した順にビットを評価して、それがsigパスで処理すべきデータであればsigパスで処理し、そうでなければ次のビットに進む。

## 【0011】

コードブロック処理が終了すると、次にrefパスとして順番に評価して行く。最後にclnパスを処理する。このとき、あるビットが処理順序の早い(優先度が高い)コーディングパスで処理されると、そのビットは他のコーディングパスでは処理しない。

## 【0012】

従って、各ビットは上記3種類の内、1種類だけのコーディングパスで処理されることになる。この処理をMSBからLSBまで繰り返すが、MSBに限りclnパスのみで処理する。

## 【0013】

例えば、 $N = 8$ ビットのとき、 $(7 \text{ ビットプレーン} \times 3 \text{ コーディングパス} + 1 \text{ コーディングパス}) \times (8 \times 12)$  コードブロックサイズのビットモデリングの演算が必要になる。

## 【0014】

## 【発明が解決しようとする課題】

上記のような従来のJPEG2000のビットモデリングの処理方法では、例えば、あるコードブロックをビットモデリングするとき、1個のデータは処理するビットプレーンに応じて $(N - 1) \times 3 + 1$  倍の処理が必要になる。

## 【0015】

これは1個のデータが、ビット精度に係わらず1回の処理だけで済む量子化等と比べて、処理速度が遅くなるという問題がある。

## 【0016】

## 【課題を解決するための手段】

この発明に係るビットモデリングの処理方法及びそれを用いる処理回路は、J P E G 2 0 0 0 のビットモデリングのシグニフィカンスプロパゲーションデコーディングパス (Significance propagation decoding pass) の処理方法において、処理対象であるビット及び周囲のビット群の有意フラグの状態に応じて変化するデータのコンテキスト及びデシジョンと、処理対象であるビット及び周囲のビット群の符号ビットの状態に応じて変化する符号ビットのコンテキスト及びデシジョンとを同時に生成しておき、処理対象であるビットの値が1のときのみ符号ビットのコンテキスト及びデシジョンを採用し、有意フラグを更新し、処理対象であるビットの値が0のときは廃棄し、処理対象であるビットの値が1か0に係わらず、処理済フラグを更新するビットモデリングの処理回路を、同一グループの4個のビットに同時に適用し、並列に処理するものである。

## 【0017】

また、J P E G 2 0 0 0 のビットモデリングのマグニチュードリファインメントパス (Magnitude refinement pass) の処理方法において、処理対象であるビットがマグニチュードリファインメントパスで初めて処理されるかどうかの情報である有意第2ビットと処理済フラグ及び有意フラグとを参照して、処理するかどうかを判断し、マグニチュードリファインメントパスで処理する場合は、処理対象であるビットのコンテキスト及びデシジョンを生成し、処理済フラグも更新するビットモデリングの処理回路を、同一グループの4個のビットに同時に適用し、並列に処理するものである。

## 【0018】

また、J P E G 2 0 0 0 のビットモデリングのクリーンアップパス (Cleanup pass) の処理方法において、処理対象である同一グループのビットが全て未処理のとき、まとめて処理できるかどうかを判断し、同一グループのビット群が全て非有意のときは特別なコンテキスト及びデシジョンを生成するビットモデリングの第1の処理回路と、処理済のビットについては何も行わず、非有

意のビットについて処理を行うビットモデリングの第2の処理回路とを設け、第1の処理回路の1つと、同一グループの4個のビットに第2の処理回路とを同時に適用し、並列に処理するものである。

## 【 0 0 1 9 】

また、J P E G 2 0 0 0 のビットモデリングの処理方法において、同一ビットプレーンについて、3種類のコーディングパスを順番に処理するものである。

## 【 0 0 2 0 】

また、J P E G 2 0 0 0 のビットモデリングの処理方法において、同一ビットプレーンの隣り合った3個のグループを、3種類のコーディングパスごとに並列に処理するものである。

## 【 0 0 2 1 】

また、請求項5記載のビットモデリングの処理方法において、同一ビットプレーンで複数並列に処理するものである。

## 【 0 0 2 2 】

また、請求項5または6記載のいずれかのビットモデリングの処理方法において、複数のビットプレーンで並列に処理するものである。

## 【 0 0 2 3 】

また、請求項1記載のビットモデリングの処理方法を用いる処理回路において、処理対象であるビットのデータの値を格納するレジスタと、処理対象であるビット及び周囲のビット群の有意フラグ及び符号ビットを格納するレジスタと、処理対象であるビットの処理済フラグを格納するレジスタとを備えるものである。

## 【 0 0 2 4 】

また、請求項2記載のビットモデリングの処理方法を用いる処理回路において、処理対象であるビットのデータの値を格納するレジスタと、処理対象であるビット及び周囲のビット群の有意フラグを格納するレジスタと、処理対象であるビットがマグニチュードリファインメントパスで初めて処理されるかどうかの情報である有意第2ビットを格納するレジスタとを備えるものである。

## 【 0 0 2 5 】

また、請求項5、6、8記載のいずれかのビットモデリングの処理方法を用い

る処理回路において、コードブロックサイズ分のデータビット、符号ビット、処理済フラグ、有意フラグ及び有意第2ビットとを格納するレジスタを備えるものである。

#### 【0026】

さらに、請求項5、6、8記載のいずれかのビットモデリングの処理方法を用いる処理回路において、データビット、符号ビット、処理済フラグ、有意フラグ及び有意第2ビットとを処理の対象とするビット分だけを格納するレジスタを備えるものである。

#### 【0027】

#### 【発明の実施の形態】

#### 実施の形態1.

以下、この発明について説明する。図1は実施の形態1によるJPEG2000のあるビットプレーンにおけるグループのビット群〇0～〇3を処理する場合を示した図である。図1を参照して、ビット群〇0～〇3で示すグループをsigパスで処理する場合、その周囲のビットの内、×で示すビットについての情報が必要になる。

#### 【0028】

図2は図1における〇0のビットを処理する場合に必要なビットを示した図である。図2を参照して、×0、×1、×2、×3、×5については同一コーディングパスの処理においてビット〇0より先に処理されるビットを示している。△4、△6、△7についてはビット〇0より後に処理されるビットを示している。

#### 【0029】

sigパスにおいてビットが処理対象となる条件は、そのビットがinsignificance（以下、非有意と示す。）であり、周囲のビットが1個以上significance（以下、有意と示す。）であることである。

#### 【0030】

従って、処理対象であるビット及び周囲のビットが有意であるかどうか（以下、有意フラグと示す。）の情報が必要になる。有意フラグはレジスタに格納して

おく。

【0031】

ビット00が非有意で、周囲のビット群 $\times 0 \sim \Delta 7$ の内、1個以上が有意であればsigパスで処理することとなる。

【0032】

周囲のビット群 $\times 0 \sim \Delta 7$ の有意フラグの状態に応じてコンテキスト（以下、データのコンテキストと示す。）を生成し、ビット00の値そのものをデシジョン（以下、データのデシジョンと示す。）とする。

【0033】

ビット00の値が1であれば、そのデータが非有意から有意になるので、有意フラグを更新して、さらに符号ビットについてもビットモデリングを行う。

【0034】

符号ビット処理には、ビット00の符号ビット及び周囲のビット群 $\times 0 \sim \Delta 7$ の符号ビットが必要になる。

【0035】

周囲のビット群 $\times 0 \sim \Delta 7$ の符号ビットの状態に応じてコンテキスト（以下、符号ビットのコンテキストと示す。）を生成し、ビット00の符号ビット及びコンテキストに応じた値のXOR演算の結果をデシジョン（以下、符号ビットのデシジョンと示す。）とする。

【0036】

データのコンテキスト／デシジョン及び符号ビットのコンテキスト／デシジョンは同時に生成しておき、必要なとき（ビット00の値が1）のみ符号ビットのコンテキスト／デシジョンを採用する。

【0037】

また、必要のないとき（ビット00の値が0）は廃棄すれば並列に処理することができる。

【0038】

また、ビット00の値が1であるか0であるかに係わらず、sigパスで処理されたときは、そのビットプレーンにおいて処理済であるという情報（以下、処

理済フラグと示す。)を更新する。

【0039】

あるビットをsigパスで処理するときは、そのビットの値、そのビットと周囲のビットの有意フラグ及び符号ビットとを参照する。

【0040】

即ち、ビット00の値、ビット00と周囲のビット群 $\times 0 \sim \Delta 7$ の有意フラグ及び符号ビットとを格納するレジスタが必要になる。

【0041】

また、ビット00の処理済フラグのためのレジスタについても用意する必要がある。

【0042】

図3は図1における01のビットを処理する場合に必要なビットを示した図である。図3を参照して、ビット01をsigパスで処理するかどうかの判断には、周囲のビット群 $\times 0 \sim \Delta 7$ の有意フラグを参照する。

【0043】

また、ビット $\times 1$ の有意フラグは図2のビット00の有意フラグと同等なので、ビット00の処理時に更新される可能性があり、ビット00の処理が終了するまでビット01の処理は開始できないはずであるが、ビット01を処理するときにビット $\times 1$ の有意フラグが更新されるかどうか解っていれば、同時に処理することが可能となる。

【0044】

即ち、ビット $\times 1$ が非有意のとき、その値が1でありその周囲のビット群が1個以上有意である場合に、直前のsigパスの処理で有意になると判断し、ビット $\times 1$ 及びその周囲のビット群の有意フラグ及びそのビットの値とを同時に参照して更新後の有意フラグとすることで、ビット00及びビット01とを同時に処理する。

【0045】

図4は図1における02のビットを処理する場合に必要なビットを示した図である。図4を参照して、ビット02をsigパスで処理するかどうかの判断

には、周囲のビット群×0～Δ7の有意フラグを参照する。

【0046】

ビット×1の有意フラグは図3のビット○1の有意フラグと同等なので、ビット○1の処理時に更新される可能性があり、ビット○1の処理が終了するまでビット○2の処理は開始できないはずであるが、ビット○2を処理するときにはビット×1の有意フラグが更新されるかどうか解っていれば、同時に処理することが可能となる。

【0047】

即ち、ビット×1が非有意のとき、その値が1でありその周囲のビット群が1個以上有意である場合に、直前のsigパスの処理で有意になると判断し、ビット×1及びその周囲のビット群の有意フラグ及びそのビットの値とを同時に参照して更新後の有意フラグとすることで、ビット○1及びビット○2とを同時に処理する。

【0048】

図5は図1における○3のビットを処理する場合に必要なビットを示した図である。図5を参照して、ビット○3をsigパスで処理するかどうかの判断には、周囲のビット群×0～Δ7の有意フラグを参照する。

【0049】

ビット×1の有意フラグは図4のビット○2の有意フラグと同等なので、ビット○2の処理時に更新される可能性があり、ビット○2の処理が終了するまでビット○3の処理は開始できないはずであるが、ビット○3を処理するときにはビット×1の有意フラグが更新されるかどうか解っていれば、同時に処理することが可能となる。

【0050】

即ち、ビット×1が非有意のとき、その値が1でありその周囲のビット群が1個以上有意である場合に、直前のsigパスの処理で有意になると判断し、ビット×1及びその周囲のビット群の有意フラグ及びそのビットの値とを同時に参照して更新後の有意フラグとすることで、ビット○2及びビット○3とを同時に処理する。

## 【0051】

この実施の形態1によると、ビット群〇0～〇3のグループの並列処理が可能となり、1ビットのデータについてはs i gパスで処理するかどうかも判断し、コンテキスト／デシジョンのセットを0／1／2個出力する回路を構成することが可能となる。

## 【0052】

また、1ビットのデータを処理する回路を4個並列に並べてそれぞれビット〇0、〇1、〇2、〇3について処理することで、1個のグループ（4個のデータ）について、0～8個のコンテキスト／デシジョンのセットを出力する回路を構成することが可能となる。

## 【0053】

実施の形態2.

次に、実施の形態2による並列処理方法について述べる。これは、図1～5をr e fパスで処理する実施の形態であるため、実施の形態1における図1～5と同様の図面を用いて説明する。

## 【0054】

まず、図1のビット群〇0～〇3で示すグループをr e fパスで処理するとき、その周囲のグループの内、×で示すビットについての情報が必要になる。このグループをさらに各ビット毎に分けると図2～5に示す各ビット〇0、〇1、〇2、〇3について、周囲8ビットの情報が必要となる。

## 【0055】

r e fパスでのビットの処理条件は、そのビットがs i gパスで処理されてなく、有意であることが必要である。従って、処理対象であるビットが処理済であるか、また、有意であるかどうかの情報が必要となる。

## 【0056】

図2のビット〇0において、未処理で有意であればr e fパスにより処理することになる。初めてビット〇0がr e fパスで処理されるとき、周囲のビット群×0～△7の有意フラグの状態に応じてデータのコンテキストを生成し、そうでないときは有意フラグの状態によらず固定のデータのコンテキストを生成する。

## 【 0 0 5 7 】

いずれのときもビット 0 の値そのものをデータのデシジョンとし、*ref* パスで処理されるときは処理済フラグを更新し、有意フラグ、処理済フラグは *sig* パスと同様のものを使用する。さらに、初めてビット 0 が *ref* パスで処理されるかどうかの情報（以下、有意第 2 ビットと示す。）はレジスタに格納しておく。

## 【 0 0 5 8 】

図 3 においてビット 1 を *ref* パスで処理するかどうかの判断は、ビット 0 の処理済フラグ及び有意フラグとを参照する。同様に、図 4、5 についてもそれぞれビット 2、ビット 3 の処理済フラグ及び有意フラグとを参照する。

## 【 0 0 5 9 】

また、図 2～5 に示す各ビット 0、1、2、3 を処理するには図 1 に示すビット群 0～3 の値、処理済フラグ及び有意フラグと、ビット 0 及びビット  $\times$  との有意フラグが必要になる。

## 【 0 0 6 0 】

この実施の形態 2 によると、*ref* パスは有意フラグを更新することがないので、ビット群 0～3 のグループの並列処理が可能となり、1 ビットのデータについては *ref* パスで処理するかどうかも判断し、コンテキスト／デシジョンのセットを 0 / 1 個出力する回路を構成することが可能となる。

## 【 0 0 6 1 】

また、1 ビットのデータを処理する回路を 4 個並列に並べてそれぞれビット 0、1、2、3 について処理することで、1 個のグループ（4 個のデータ）について、0～4 個のコンテキスト／デシジョンのセットを出力する回路を構成することが可能となる。

## 【 0 0 6 2 】

実施の形態 3.

実施の形態 3 による並列処理方法について述べる。これは、図 1～5 を *cln* パスで処理する実施の形態であるため、実施の形態 1 における図 1～5 と同様の図面を用いて説明する。

## 【 0 0 6 3 】

まず、図 1 のビット群 0 0 ~ 0 3 で示すグループを c l n パスで処理するとき、その周囲のグループの内、×で示すビットについての情報が必要になる。このグループをさらに各ビット毎に分けると図 2 ~ 5 に示す各ビット 0 0, 0 1, 0 2, 0 3 について、周囲 8 ビットの情報が必要となる。

## 【 0 0 6 4 】

c l n パスでのビットの処理条件は、そのビットが未処理であることである。但し、MSB では c l n パスしか処理しないので全てのビットが処理対象となる。従って、処理対象であるビットが処理済であるかどうかの情報が必要となる。

## 【 0 0 6 5 】

また、c l n パスでは図 2 ~ 5 に示す各ビット 0 0, 0 1, 0 2, 0 3 において、全て未処理のとき、まとめて処理できるかどうかを判断する。図 2 ~ 5 に示す各ビット 0 0, 0 1, 0 2, 0 3 のそれぞれにおいて、周囲のビット群 × 0 ~ Δ 7 が全て非有意のときは特別なコンテキスト（以下、ランレングスのコンテキストと示す。）を生成する。

## 【 0 0 6 6 】

また、同時に図 2 ~ 5 に示す各ビット 0 0, 0 1, 0 2, 0 3 の値が全て 0 であればデシジョン（以下、ランレングスのデシジョンと示す。） 0 を生成し、そのグループの処理を終了する。

## 【 0 0 6 7 】

また、図 2 ~ 5 に示す各ビット 0 0, 0 1, 0 2, 0 3 の値が 1 つでも 1 であれば、ランレングスのデシジョンは 1 とする。このとき図 2 ~ 5 に示す各ビット 0 0, 0 1, 0 2, 0 3 の内、最初に 1 の値を持つビットの位置を 2 ビットのデータで表す。

## 【 0 0 6 8 】

即ち、ビット 0 0 が最初の 1 であれば 0 0、ビット 0 1 であれば 0 1、以下、同様 1 0、1 1 とする。これらの 2 つのデータはランレングスに続くコンテキスト（以下、UNIFORM のコンテキストと示す。）とともに生成され、それぞれデシジョン（以下、UNIFORM のデシジョンと示す。）として出力される

## 【 0 0 6 9 】

ランレングスのコンテキストだけが出力されるときは、全て非有意のままであるので有意フラグの更新は行われぬ。

## 【 0 0 7 0 】

UNIFORMのコンテキストも出力されるとき、UNIFORMのデシジョンが示す位置のビットは非有意から有意になるので、そのビットの有意フラグを更新する。

## 【 0 0 7 1 】

但し、UNIFORMのコンテキスト自体が、そのデシジョンの示す位置のビットが1であることを表すので、sigパスと同様にそのビットの符号ビットのコンテキストを生成する。

## 【 0 0 7 2 】

clnパスでの符号ビットのコンテキストの生成方法はsigパスと同様であり、UNIFORMのデシジョンが11のときは符号ビットのみの処理で終了となるが、それ以外のときは残るビットを1ビットづつ処理する。

## 【 0 0 7 3 】

残るビットの処理の方法はそのビットの周囲のビットが全て非有意でも処理することを除いて、sigパスと同様である。また、UNIFORMのデシジョンが示す位置より前のビットは非有意のままであるので、有意フラグの更新は行わぬ。

## 【 0 0 7 4 】

この実施の形態3によると、ランレングスのコンテキストの処理対象とならないときは、そのグループ中のビットの内、未処理のものを順番に処理する。また、UNIFORMのコンテキスト以後のビット、またはランレングスのコンテキストの処理対象とならないときの未処理のビットについては、sigパスと同様の処理を行うためsigパスと同様の回路を構成することが可能となる。

## 【 0 0 7 5 】

また、sigパスでは各ビットが独立したものを並列に処理することで0～8

個のコンテキスト／デシジョンのセットを出力する回路を構成しているが、c l nパスによると、さらに0ランレングスのコンテキスト／デシジョン、UNIFORMのコンテキスト／デシジョンを処理する回路を並列に処理させ、

- ・ 0ランレングスのコンテキストを含む場合
  - UNIFORMのコンテキストを含むか
  - UNIFORMのコンテキスト以後のビット数
- ・ 0ランレングスのコンテキストを含まない場合
  - c l nパスで処理するビット数

を考慮して、0ランレングス、UNIFORM、各ビットのコンテキスト／デシジョンを計算することで0～10個のコンテキスト／デシジョンのセットを出力する回路を構成することが可能となる。

【0076】

実施の形態4.

実施の形態1～3については、各コーディングパスについて個別の回路構成について述べたものであるが、実施の形態4では同一ビットプレーン内の同一グループについて実施の形態1～3の回路で同時に処理するものである。

【0077】

この場合、3種類のコーディングパスで情報を共有する必要があるため、コードブロックサイズ分の有意フラグ、有意第2ビット、処理済フラグをレジスタに格納しておく必要がある。

【0078】

また、符号ビット、処理対象となるビットプレーンのデータビットについては、コードブロックサイズ分、または実際に処理対象となる部分を各コーディングパス毎に読み出す場合はその分だけレジスタに格納しておく必要がある。

【0079】

この実施の形態4によると、実施の形態1～3に比べ処理速度を速くすることができる。

【0080】

実施の形態5.

図6は実施の形態5による3個の連続したグループを処理する場合に使用するデータの範囲を示す図である。実施の形態4では、同一ビットプレーン内の同一グループを3種類のコーディングパスで同時に処理する方法であった。

【0081】

しかし、コードブロック内のあるビットプレーンについては、sigパス→refパス→clnパスの順で処理して行くが、各ビットは1個のコーディングパスだけで処理されることや、sigパスあるいはclnパス等複数のパスで有意フラグを更新すること等の制約により、正確なビットモデリングを得ようとすれば、各コーディングパスを順番に処理する必要がある。

【0082】

つまり、あるビットプレーンにおいて、4個のビットを同時に処理しているとはいえ実際には3種類のコーディングパスで順番に処理することになり、回路としての動作周波数を低く押さえる必要がある。これを解決するために実施の形態5では、横方向に連続したグループ（ビット群00～03、ビット群04～07、ビット群08～011）をそれぞれ別のコーディングパスで処理する方法をとっている。

【0083】

図7はビット群08～011のグループを処理する場合に使用するデータの範囲を示す図である。また、図8はビット群04～07のグループを処理する場合に使用するデータの範囲を示す図である。また、図9はビット群00～03のグループを処理する場合に使用するデータの範囲を示す図である。

【0084】

sigパスは非有意であり、かつ周囲のビットに1個以上有意であるビットが存在するときに処理するコーディングパスであり、最初に実行され、他の2個のパスからは何の制約も受けずに処理することができる。

【0085】

図6で示す3個のグループではビット群08～011を処理するコーディングパスであり、図7がそのデータの使用範囲である。ビット群08～011をsigパスで処理するので、条件によりビット群08～011の処理済フラグ、有意

フラグが更新される。

【0086】

図6～9におけるビット群 $\times 5 \sim \times 8$ 、ビット群 $\bigcirc 0 \sim \bigcirc 3$ 、ビット群 $\bigcirc 4 \sim \bigcirc 7$ はsigパスにより既に処理済である。

【0087】

次に、refパスは既に有意であるデータ进行处理するコーディングパスであり、処理時に有意であるかどうかを考慮する必要がある。

【0088】

そのビットが既に有意であるかどうかは有意フラグを参照するが、そのビットが同じビットプレーンのsigパスで処理され有意になったときは処理できない。

【0089】

即ち、sigパスで処理されたかどうかを判断する必要があるため、sigパスと同時に処理することはできない。

【0090】

図6で示す3個のグループではビット群 $\bigcirc 4 \sim \bigcirc 7$ 进行处理するコーディングパスであり、図8がそのデータの使用範囲である。ビット群 $\bigcirc 4 \sim \bigcirc 7$ をrefパスで処理するので、ビット群 $\bigcirc 4 \sim \bigcirc 7$ の処理済フラグ、有意フラグを参照し、条件により処理済フラグを更新する。

【0091】

sigパスでビット群 $\bigcirc 8 \sim \bigcirc 11$ 、refパスでビット群 $\bigcirc 4 \sim \bigcirc 7$ 进行处理することで、refパスではsigパスで更新した後のビット群 $\bigcirc 4 \sim \bigcirc 7$ のフラグを参照することができる。

【0092】

また、この実施の形態5では、回路規模を最小とするためsigパスとrefパスで処理するグループは横方向に1グループずらした構成を示したが、2グループ以上ずらしてもよい。

【0093】

図6、8、9におけるビット群 $5 \times \sim \times 8$ 、ビット群 $\bigcirc 0 \sim \bigcirc 3$ はrefパス

により既に処理済である。

【0094】

また、c l nパスは非有意であり、s i gパスで処理されなかったデータを処理するコーディングパスである。この実施の形態5では、有意なデータはr e fパスで既に処理されており、非有意なデータの内s i gパスで処理されなかったデータというのは、未処理のデータ全てであるので、処理済フラグを参照し、未処理のデータのみを処理する。

【0095】

処理対象となるビット及び周囲ビットの有意フラグを参照する必要があるため、s i gパスと同時に処理することはできない。

【0096】

図6で示す3個のグループではビット群〇0～〇3を処理するコーディングパスであり、図9がそのデータの使用範囲である。ビット群〇0～〇3をc l nパスで処理するので、ビット群〇4～〇7の有意フラグ、ビット群〇0～〇3の処理済フラグを参照し、条件により有意フラグを更新する。

【0097】

また、図6、9におけるビット群×5～×8はc l nパスにより既に処理済である。

【0098】

s i gパスでビット群〇8～〇11、r e fパスでビット群〇4～〇7、c l nパスでビット群〇0～〇3を処理することで、c l nパスではs i gパス及びr e fパスで更新した後のビット群〇0～〇3、ビット群〇4～〇7のフラグを参照することができる。

【0099】

また、この実施の形態5では、回路規模を最小とするためs i gパスとr e fパスで処理するグループは横方向に1グループずらした構成を示したが、s i gパス、r e fパス及びc l nパスで処理するグループ数は任意でよい。

【0100】

上記構成で3種類のコーディングパスを並列に処理するとき、実施の形態1～

4と同様の方法とあわせて、4個のデータを3種類のコーディングパスだけ並列に処理できる。

#### 【0101】

また、図6においてsigパスでビット群 $\Delta 9 \sim \Delta 12$ 、refパスでビット群 $\bigcirc 8 \sim \bigcirc 11$ 、clnパスでビット群 $\bigcirc 4 \sim \bigcirc 7$ というように処理するグループをずらすことで連続に処理することができる。

#### 【0102】

各コーディングパスは0～10個のコンテキスト／デシジョンのセットを出力するが、これらのデータはコーディングパス毎に格納する必要がある。

#### 【0103】

この実施の形態5によると、コードブロックサイズ分の有意フラグ、有意第2ビットの情報をレジスタに格納しておく必要があるが、3種類のコーディングパスを1度に処理するため、符号ビット、処理対象となるビットプレーンのデータビット、処理済フラグについては実際に処理する場合に処理の対象となっている部分だけレジスタに格納しておけばよい。

#### 【0104】

また、図6のように、各コーディングパスで処理するグループの間隔を1グループにした場合、処理の対象となっている部分というのは、符号ビットでは $\bigcirc \Delta \times$ の30ビット分、データビット及び処理済フラグでは $\times 5 \sim \times 8$ 、 $\bigcirc 0 \sim \bigcirc 11$ 、 $\Delta 9 \sim \Delta 12$ の20ビット分になる。

#### 【0105】

従って、実施の形態1～4に比べ、高速に処理でき、回路規模を縮小できる。

#### 【0106】

実施の形態6.

次に、実施の形態6による並列処理方法について述べる。これは、図6における有意フラグ及び有意第2ビットを処理の対象とする部分（ $\bigcirc \Delta \times$ の30ビット分（有意フラグ）、 $\times 5 \sim \times 8$ 、 $\bigcirc 0 \sim \bigcirc 11$ 、 $\Delta 9 \sim \Delta 12$ の20ビット分（有意第2ビット））だけのレジスタ量で実行する方法である。

#### 【0107】

有意フラグがコードブロックサイズ分必要であるのは、MSBからLSBまでビットモデリングをしている間、情報を共有する必要があるため、保持する必要があるからである。

#### 【0108】

しかし、あるビットプレーンの処理においては、その直前のビットプレーンでの処理後の状態が解ればよく、直前のビットプレーンまで処理した後の状態を予め計算することができれば、ビットプレーン間で情報を共有する必要性がなくなる。

#### 【0109】

実施の形態6では、処理しようとするデータの値を調べ、処理対象となるビットプレーンより上位に値が1であるビットがあるかどうかで判断する方法をとる。

#### 【0110】

図10は、ビットモデリングで処理しようとするある1つのデータを符号ビット、及び絶対値に分け、その値を示した図である。図10を参照して、MSBから処理対象であるビットプレーンより1個上のビットまでの値は全て0であるので、このデータのsigパス処理前の有意フラグは非有意とする。

#### 【0111】

また、図11では、MSBから処理対象であるビットプレーンより1個上のビットまでの値が1であるので、このデータのsigパス処理前の有意フラグは有意とする。

#### 【0112】

これをビットプレーン毎にデータビットを読み出すときに実行すれば、ビットプレーン間で情報を共有する必要性がなくなるため、有意フラグのレジスタ数を削減することができる。

#### 【0113】

従って、有意フラグは図6に示す○△×の30ビット分だけレジスタを持てばよいことになる。

#### 【0114】

また、有意第2ビットも同様に、初めてrefパスで処理するビットプレーン、即ち、有意になった次のビットプレーンを予め計算しておくことにより、ビットプレーン間で情報を共有する必要がなくなる。

## 【0115】

この実施の形態6では、処理しようとするデータを調べ、処理対象となるビットプレーンより1ビット上位の値が1で、それより上位のビットが全て0であるかどうかで判断する。

## 【0116】

図10を例にとると、処理対象のビットプレーンより上位のビットの値が全て0であるので有意第2ビットにはならない。また、図11では、処理対象するビットプレーンの1ビット上位の値が1で、それより上位ビットの値が全て0であるので有意第2ビットになる。

## 【0117】

これをビットプレーン毎にデータビットを読み出すときに実行すれば、ビットプレーン間で情報を共有する必要がなくなるため、有意第2ビットのレジスタ数を削減することができる。

## 【0118】

従って、有意第2ビットは図6に示す×5～×8、○0～○11、△9～△12の20ビット分だけ持てばよいことになる。

## 【0119】

実施の形態6は、同一のビットプレーンにおいて3種類のコーディングパスを並列に処理する回路であり、以下の実施の形態の基本回路とする。

## 【0120】

実施の形態7.

この実施の形態7は、同一のビットプレーンにおいて実施の形態6の回路を複数使用する処理方法である。図13は3個の連続したグループを複数並列に処理する場合に使用するデータの範囲を示す図である。これはビット群○000～○011、○100～○111について2つの実施の形態6で述べた回路を使用するものである。

## 【 0 1 2 1 】

図 1 3 を参照して、ビット群 0 0 0 0 ～ 0 0 1 1 については実施の形態 6 と同様の回路を使用してそのまま処理する。また、ビット群  $\times 1 0 0 \sim \times 1 0 4$  は本来ビット群 0 1 0 0 ～ 0 1 1 1 よりも前に処理されるべきデータであるので有意フラグを予め計算しておく必要がある。

## 【 0 1 2 2 】

実施の形態 6 では、処理対象となるビットプレーンの 1 ビット上位のビットプレーンまでの有意フラグの計算方法を示したが、この実施の形態 7 では、さらに縦方向に 1 段上のグループの処理後の情報が必要になる。これは実施の形態 6 で示した方法に加えて、処理対象のビットプレーンのデータも考慮することで計算できる。

## 【 0 1 2 3 】

以下に、図 1 3 の  $\times 1 0 0$  が、図 1 0 ～ 1 2 のような値を持っている場合について述べる。

## 【 0 1 2 4 】

図 1 0 では、MSB から処理対象となるビットプレーンまで値が全て 0 なので非有意である。図 1 1 では、MSB から処理対象となるビットプレーンの 1 ビット上位までに値が 1 であるビットがあるので有意である。

## 【 0 1 2 5 】

また、図 1 2 では、MSB から処理対象となるビットプレーンまでに値が 1 であるビット（この場合は、処理対象のビットプレーン）があるので有意である。このように、縦方向に 1 段上のグループまで、有意フラグを計算することが可能となる。

## 【 0 1 2 6 】

但し、ビット群  $\times 1 0 5 \sim \times 1 0 8$ 、ビット群  $\Delta 1 0 9 \sim \Delta 1 1 7$  については実施の形態 1 ～ 6 と同様の処理とする。例えば、ビット  $\Delta 1 1 7$  が図 1 2 のような値を持っていた場合、非有意となる。

## 【 0 1 2 7 】

この実施の形態 7 によると、同一のビットプレーンにおいて複数のグループの

並列処理が可能となる。また、縦方向に連続したグループを処理する例を示したが、縦方向のグループの間隔は1に限らず任意でよく、並列に処理する実施の形態6と同様の回路も2個以上の任意の数でよい。尚、並列回路を2個設ければ、回路規模は2倍となるが、処理性能も2倍となる。

## 【0128】

実施の形態8.

さらに、実施の形態8では異なるビットプレーンでの並列処理が可能な方法について述べる（図示せず。）。これは実施の形態6で述べた方法を異なるビットプレーンに対して同時に使用することで、複数のビットプレーン（2以上の任意の数）での並列処理が可能となる。

## 【0129】

また、この実施の形態8も並列回路を2個設ければ、回路規模は2倍となるが、処理性能も2倍となる。

## 【0130】

さらに、実施の形態7と同様の回路を異なるビットプレーンに対して同時に実行させることも可能である。尚、この場合も任意の数のビットプレーンでの並列処理が可能である。

## 【0131】

## 【発明の効果】

この発明に係るビットモデリングの処理方法及びそれを用いる処理回路は、J P E G 2 0 0 0 のビットモデリングのシグニフィカンスプロパゲーションデコーディングパス（Significance propagation decoding pass）の処理方法において、処理対象であるビット及び周囲のビット群の有意フラグの状態に応じて変化するデータのコンテキスト及びデシジョンと、処理対象であるビット及び周囲のビット群の符号ビットの状態に応じて変化する符号ビットのコンテキスト及びデシジョンとを同時に生成しておき、処理対象であるビットの値が1のときのみ符号ビットのコンテキスト及びデシジョンを採用し、有意フラグを更新し、処理対象であるビットの値が0のときは廃棄し、処理対象であるビットの値が1か0に係わらず、処理済フラグを更新するビ

ットモデリングの処理回路を、同一グループの4個のビットに同時に適用し、並列に処理するので、処理速度を速くすることができる。

## 【 0 1 3 2 】

また、J P E G 2 0 0 0 のビットモデリングのマグニチュードリファインメントパス (M a g n i t u d e   r e f i n e m e n t   p a s s) の処理方法において、処理対象であるビットがマグニチュードリファインメントパスで初めて処理されるかどうかの情報である有意第2ビットと処理済フラグ及び有意フラグとを参照して、処理するかどうかを判断し、マグニチュードリファインメントパスで処理する場合は、処理対象であるビットのコンテキスト及びデシジョンを生成し、処理済フラグも更新するビットモデリングの処理回路を、同一グループの4個のビットに同時に適用し、並列に処理するので、さらに処理速度を速くすることができる。

## 【 0 1 3 3 】

また、J P E G 2 0 0 0 のビットモデリングのクリーンアップパス (C l e a n u p   p a s s) の処理方法において、処理対象である同一グループのビットが全て未処理のとき、まとめて処理できるかどうかを判断し、同一グループのビット群が全て非有意のときは特別なコンテキスト及びデシジョンを生成するビットモデリングの第1の処理回路と、処理済のビットについては何も行わず、非有意のビットについて処理を行うビットモデリングの第2の処理回路とを設け、第1の処理回路の1つと、同一グループの4個のビットに第2の処理回路とを同時に適用し、並列に処理するので、さらに処理速度を速くすることができる。

## 【 0 1 3 4 】

また、J P E G 2 0 0 0 のビットモデリングの処理方法において、同一ビットプレーンについて、3種類のコーディングパスを順番に処理するので、さらに処理速度を速くすることができる。

## 【 0 1 3 5 】

また、J P E G 2 0 0 0 のビットモデリングの処理方法において、同一ビットプレーンの隣り合った3個のグループを、3種類のコーディングパスごとに並列に処理するので、さらに処理速度を速くすることができ、回路規模を縮小するこ

とができる。

【 0 1 3 6 】

また、請求項 5 記載のビットモデリングの処理方法において、同一ビットプレーンで複数並列に処理するので、さらに処理速度を速くすることができ、回路規模を縮小することができる。

【 0 1 3 7 】

また、請求項 5 または 6 記載のいずれかのビットモデリングの処理方法において、複数のビットプレーンで並列に処理するので、さらに処理速度を速くすることができる。

【 0 1 3 8 】

また、請求項 1 記載のビットモデリングの処理方法を用いる処理回路において、処理対象であるビットのデータの値を格納するレジスタと、処理対象であるビット及び周囲のビット群の有意フラグ及び符号ビットを格納するレジスタと、処理対象であるビットの処理済フラグを格納するレジスタとを備えるので、処理速度を速くすることができる。

【 0 1 3 9 】

また、請求項 2 記載のビットモデリングの処理方法を用いる処理回路において、処理対象であるビットのデータの値を格納するレジスタと、処理対象であるビット及び周囲のビット群の有意フラグを格納するレジスタと、処理対象であるビットがマグニチュードリファインメントパスで初めて処理されるかどうかの情報である有意第 2 ビットを格納するレジスタとを備えるので、さらに処理速度を速くすることができる。

【 0 1 4 0 】

また、請求項 5、6、8 記載のいずれかのビットモデリングの処理方法を用いる処理回路において、コードブロックサイズ分のデータビット、符号ビット、処理済フラグ、有意フラグ及び有意第 2 ビットとを格納するレジスタを備えるので、処理速度を速くすることができ、回路規模を縮小することができる。

【 0 1 4 1 】

さらに、請求項 5、6、8 記載のいずれかのビットモデリングの処理方法を用

いる処理回路において、データビット、符号ビット、処理済フラグ、有意フラグ及び有意第2ビットとを処理の対象とするビット分だけを格納するレジスタを備えるので、さらに処理速度を速くすることができ、回路規模を縮小することができる。

【図面の簡単な説明】

【図1】 この発明の実施の形態1によるJ P E G 2 0 0 0のビットモデリングにおいて、あるグループを処理する場合に使用するデータの範囲を示す図である。

【図2】 この発明の実施の形態1によるJ P E G 2 0 0 0のビットモデリングにおいて、ビット00を処理する場合に使用するデータの範囲を示す図である。

【図3】 この発明の実施の形態1によるJ P E G 2 0 0 0のビットモデリングにおいて、ビット01を処理する場合に使用するデータの範囲を示す図である。

【図4】 この発明の実施の形態1によるJ P E G 2 0 0 0のビットモデリングにおいて、ビット02を処理する場合に使用するデータの範囲を示す図である。

【図5】 この発明の実施の形態1によるJ P E G 2 0 0 0のビットモデリングにおいて、ビット03を処理する場合に使用するデータの範囲を示す図である。

【図6】 この発明の実施の形態5による3個の連続したグループを処理する場合に使用するデータの範囲を示す図である。

【図7】 この発明の実施の形態5によるビット群08～011のグループを処理する場合に使用するデータの範囲を示す図である。

【図8】 この発明の実施の形態5によるビット群04～07のグループを処理する場合に使用するデータの範囲を示す図である。

【図9】 この発明の実施の形態5によるビット群00～03のグループを処理する場合に使用するデータの範囲を示す図である。

【図10】 この発明の実施の形態6によるJ P E G 2 0 0 0のビットモデ

リングにおいて、処理の対象となるデータを符号ビットとデータに分けてビットプレーン毎に値を示した図である。

【図 1 1】 この発明の実施の形態 6 による J P E G 2 0 0 0 のビットモデリングにおいて、別の処理の対象となるデータを符号ビットとデータに分けてビットプレーン毎に値を示した図である。

【図 1 2】 この発明の実施の形態 6 による J P E G 2 0 0 0 のビットモデリングにおいて、さらに別の処理の対象となるデータを符号ビットとデータに分けてビットプレーン毎に値を示した図である。

【図 1 3】 この発明の実施の形態 7 による 3 個の連続したグループを複数並列に処理する場合に使用するデータの範囲を示す図である。

【図 1 4】 従来の J P E G 2 0 0 0 のビットモデリングにおいて、ビットプレーン内で処理するグループの順番を示す図である。

【図 1 5】 従来の J P E G 2 0 0 0 のビットモデリングにおいて、別のビットプレーン内で処理するグループの順番を示す図である。

【符号の説明】

○ 0 ~ ○ 3、○ 4 ~ ○ 7、○ 8 ~ ○ 1 1      ビット群

× 0 ~ Δ 7      周囲のビット群

○ 0、○ 1、○ 2、○ 3      各ビット

× 5 ~ × 8、○ 0 ~ ○ 1 1、Δ 9 ~ Δ 1 2      ビット群

○ 0 0 0 ~ ○ 0 1 1、○ 1 0 0 ~ ○ 1 1 1      ビット群

【書類名】

図面

【図 1】

	0	1	2	3	4	5	6	7
0								
1								
2								
3								
4								
5								
6								
7								
8								
9								
10								
11								

【図 2】

×0	×1	×2
×3	○0	△4
×5	△6	△7

【図 3】

×0	×1	△2
×3	○1	△4
×5	△6	△7

【図 4】

×0	×1	△2
×3	○2	△4
×5	△6	△7

【図 5】

×0	×1	△2
×3	○3	△4
△5	△6	△7

【図 6】

× 0	× 1	× 2	× 3	× 4
× 5	○ 0	○ 4	○ 8	△ 9
× 6	○ 1	○ 5	○ 9	△10
× 7	○ 2	○ 6	○10	△11
× 8	○ 3	○ 7	○11	△12
△13	△14	△15	△16	△17

【図 7】

× 3	× 4	× 5
○ 4	○ 8	△ 9
○ 5	○ 9	△10
○ 6	○10	△11
○ 7	○11	△12
△15	△16	△17

【図 8】

× 1	× 2	× 3
○ 0	○ 4	○ 8
○ 1	○ 5	○ 9
○ 2	○ 6	○10
○ 3	○ 7	○11
△14	△15	△16

【図 9】

× 0	× 1	× 3
× 5	○ 0	○ 4
× 6	○ 1	○ 5
× 7	○ 2	○ 6
× 8	○ 3	○ 7
△13	△14	△15

【図 1 0】

1

符号ビット

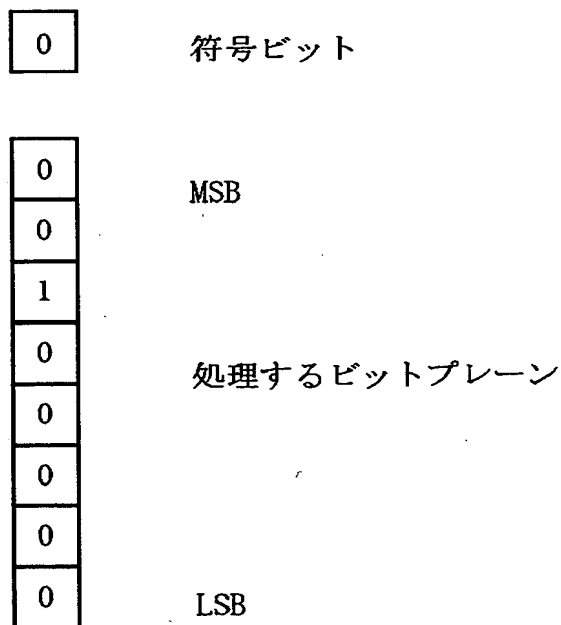
0  
0  
0  
0  
0  
1  
0  
0

MSB

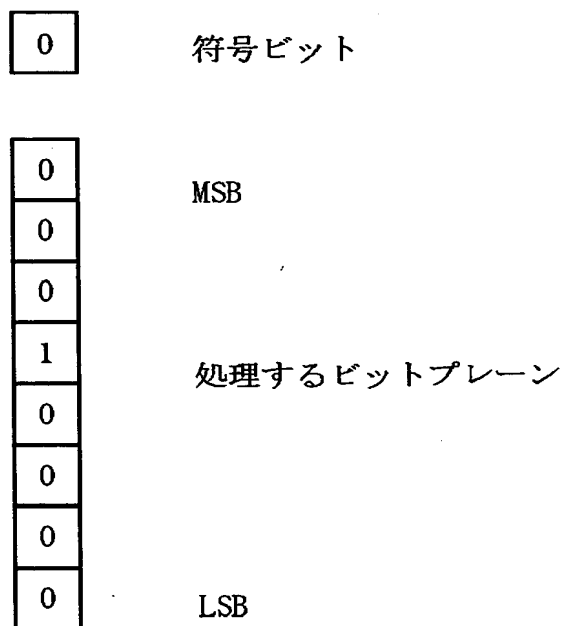
処理するビットプレーン

LSB

【図 1 1】



【図 1 2】



【図 1 3】

×000	×001	×002	×003	×004					
×005	○000	○004	○008	△009					
×006	○001	○005	○009	△010					
×007	○002	○006	○010	△011					
×008	○003	○007	○011	△012	×100	×101	×102	×103	×104
△013	△014	△015	△016	△017	×105	○100	○104	○108	△109
					×106	○101	○105	○109	△110
					×107	○102	○106	○110	△111
					×108	○103	○107	○111	△112
					△113	△114	△115	△116	△117

【図 1 4】

	0	1	2	3	4	5	6	7
0								
1								
2	0	1	2	3	4	5	6	7
3								
4								
5								
6	8	9	10	11	12	13	14	15
7								
8								
9								
10	16	17	18	19	20	21	22	23
11								

【図 1 5】

	0	1	2	3	4	5	6	7
0	0	4						
1	1	5						
2	2	6						
3	3	7						
4								
5								
6								
7								
8								
9								
10								
11								

【書類名】 要約書

【要約】

【課題】 処理速度の速いビットモデリングの処理方法及びそれを用いる処理回路を得ることである。

【解決手段】 s i g パスにより、処理対象であるビットと周囲のビット群の有意フラグの状態及び符号ビットの状態に応じて変化するコンテキスト及びデシジョンとを同時に生成し、処理対象であるビットの値が1のときのみ符号ビットのコンテキスト及びデシジョンを採用し、有意フラグを更新し、処理対象であるビットの値が0のときは廃止し、処理対象であるビットの値が1か0に係わらず、処理済フラグを更新するビットモデリングの処理回路を、同一グループのビット〇0～〇3に同時に適用し、並列に処理する。

【選択図】 図1

出 願 人 履 歴 情 報

識別番号 [000006013]

1. 変更年月日 1990年 8月24日  
[変更理由] 新規登録  
住 所 東京都千代田区丸の内2丁目2番3号  
氏 名 三菱電機株式会社